

CLAIMS

1. A method for modifying a pre-existing graphical user interface (GUI), said graphical user interface comprising a tree structure of GUI components each having a display area defined by area parameters, said graphical user interface being operable by a cursor control input device and at least some of the GUI components having listeners for responding to events from said input device, comprising the following steps:

(a) providing at least some of the components with a component mode indicator identifying the component as being in an operation mode or in an edit mode,

(b) detecting a main trigger event inputted by a user;

(c) upon occurrence of a main trigger event, inhibiting all component listeners and updating component mode indicators based on the location of the cursor relative to the component areas at the time of the main trigger event,

(d) when at least one component is in edit mode, detecting area parameter change events from the cursor controlled input device,

(e) when such area parameter change events occur, identifying from these events a target component for area parameter change and applying corresponding area parameter changes to the target component.

2. A method according to claim 1, wherein said mode indicators are defined by a component path identifying all components in edit mode.

3. A method according to claim 2, wherein said component path includes a sub-set of components, wherein two adjacent components in the path have a direct parent-to-child relationship.

4. A method according to claim 3, wherein step (c) includes the following sub-steps:

- (c1) identifying the component of lowest level in the tree structure, on which the cursor is located at the time of the main trigger event;

- (c2) if said component is contained in the component path, then removing from the component path said component as well as any descendant component thereof;

(c3) if said component is not contained in the component path, then identifying the closest ascendant component of said component which is itself contained in the component path, and adding to the component path the ascendant of said component which is a direct child of said closest ascendant.

5. A method according to claim 1, wherein said area parameter change events comprise a button depression, a cursor dragging and a button release, and wherein step (c) comprises the following sub-steps:

(e1) detecting a button depression,

(e2) upon occurrence of such button depression, identifying as a target component a component meeting the following criteria:

- the cursor is located in the area of this component at the time of button depression,

- the component is a direct child of a component which is in edit mode, and

- the component is not itself in edit mode,

(e3) changing at least one area parameter of this component according to cursor movements between button depression and button release.

6. A method according to claim 5, wherein step (e) further comprises, between sub-steps (e2) and (e3), the following sub-steps:

(e2') determining a position of the cursor at the time of button depression relative to borders of the target component,

(e2'') depending of cursor position, selecting an area parameter change among an area resizing and an area displacement.

7. A method according to claim 1, further comprising the preliminary steps of:

- establishing and storing a list of all listeners of graphical user interface components, and

-registering a device adapted to perform the method as an input device event listener for at least some of the graphical user interface components.

5

8. A logical device for modifying a pre-existing graphical user interface (GUI), said graphical user interface comprising a tree structure of GUI components each having a display area defined by area parameters, said graphical user interface being operable by a cursor control input device and at least some of the GUI components having listeners for responding to events from said input device, comprising:

10

- a mouse-type user-input listening device responsive to a main trigger event and to area parameter change events, and for identifying a current component area of the graphical

15

- a mode manager for setting and storing mode indicator data for at least some of the GUI components, wherein each component mode indicator identifies the component as being in an operation mode or in an edit mode, said mode manager being responsive to main trigger event and cursor location at the occurrence of said main trigger events for changing component mode, and

20

- a component area parameter changing device for changing at least one component area parameter of a given component of the graphical user interface in response to user inputted area parameter change events, wherein said given component is determined from said inputted area parameter change events and from said mode indicators.

25

9. A device according to claim 8, wherein said mode indicators are defined by a component path identifying all components in edit mode.

30

10. A device according to claim 9, wherein said component path includes a subset of components, wherein two adjacent components in the path have a direct parent-to-child relationship.

11. A device according to claim 3, wherein said mode manager comprises:

- a logic for identifying the component of lowest level in the tree structure, on which the cursor is located at the time of the main trigger event;

- a logic for removing from the component path said component as well as any descendant component thereof if said component is contained in the component path,

- a logic for identifying the closest ascendant component of said component which is itself contained in the component path, and adding to the component path the ascendant of said component which is a direct child of said closest ascendant, if said component is not contained in the component path.

12. A device according to claim 8, wherein said area parameter change events comprise a button depression, a cursor dragging and a button release, and wherein said area parameter changing devices is capable of:

- detecting a button depression,

- upon occurrence of such button depression, identifying as a target component a component meeting the following criteria:

* the cursor is located in the area of this component at the time of button depression,

* the component is a direct child of a component which is in edit mode, and

* the component is not itself in edit mode,

and

- changing at least one area parameter of this component according to cursor movements between button depression and button release.

13. A device according to claim 12, wherein said area parameter changing device is further capable of:

- determining a position of the cursor at the time of button depression relative to borders of the target component, and

- depending of cursor position, selecting an area parameter change among an area resizing und an area displacement.

14. A device according to claim 8, wherein it is an object oriented device and
5 wherein it can be implemented by an instantiation instruction including a GUI identifier as an argument.

FIG. 1
FIG. 2
FIG. 3
FIG. 4
FIG. 5
FIG. 6
FIG. 7
FIG. 8
FIG. 9
FIG. 10
FIG. 11
FIG. 12
FIG. 13
FIG. 14
FIG. 15
FIG. 16
FIG. 17
FIG. 18
FIG. 19
FIG. 20
FIG. 21
FIG. 22
FIG. 23
FIG. 24
FIG. 25
FIG. 26
FIG. 27
FIG. 28
FIG. 29
FIG. 30
FIG. 31
FIG. 32
FIG. 33
FIG. 34
FIG. 35
FIG. 36
FIG. 37
FIG. 38
FIG. 39
FIG. 40
FIG. 41
FIG. 42
FIG. 43
FIG. 44
FIG. 45
FIG. 46
FIG. 47
FIG. 48
FIG. 49
FIG. 50
FIG. 51
FIG. 52
FIG. 53
FIG. 54
FIG. 55
FIG. 56
FIG. 57
FIG. 58
FIG. 59
FIG. 60
FIG. 61
FIG. 62
FIG. 63
FIG. 64
FIG. 65
FIG. 66
FIG. 67
FIG. 68
FIG. 69
FIG. 70
FIG. 71
FIG. 72
FIG. 73
FIG. 74
FIG. 75
FIG. 76
FIG. 77
FIG. 78
FIG. 79
FIG. 80
FIG. 81
FIG. 82
FIG. 83
FIG. 84
FIG. 85
FIG. 86
FIG. 87
FIG. 88
FIG. 89
FIG. 90
FIG. 91
FIG. 92
FIG. 93
FIG. 94
FIG. 95
FIG. 96
FIG. 97
FIG. 98
FIG. 99
FIG. 100